Kinetic Monte Carlo and Its Implementation in 2D Supramolecular Assembly

ZHAO Heze*

Supervisor: Prof. LIN Nian

Department of Physics, Hong Kong University of Science and Technology

Clear Water Bay, Hong Kong

Abstract

In this UROP1000 report, we present a fundamental investigation into the *kinetic Monte Carlo* (KMC) method, and study one particular system of KMC simulations. The report begins with introducing the *Monte Carlo* method and the motivation for KMC. Then we walk through the basic theory and algorithm of KMC, together with the calculation of the rate constant and simplification to *lattice KMC*. More advanced topics and algorithm variants are briefly mentioned. Finally we explain the mechanism and simulation outcomes of a hands-on lattice KMC system of cross shaped molecules.

Keywords: *kinetic Monte Carlo; lattice KMC; Markov chain; the n-fold way; rate constant*

1 Introduction

Named after the Monacan ward famous for its casino, *Monte Carlo* (MC) refers to a wide range of methods sharing the central problem-solving philosophy: using random numbers. It thus has the ability to offer a numerical approach to an analytically-difficult problem. In Figure 1 and Table 1, an elementary MC algorithm was implemented.

Just like the π estimating (or any other MC) algorithm, *kinetic Monte Carlo* (KMC) simulations utilize random numbers to numerically solve equations, in this case being mainly the Markovian Master Equation (MME) of particle systems along transition of states. The dynamical evolution of such systems is important in many fields, e.g. transport (diffusion) [1], reaction kinetics [2], and crystal growth [3]. One example of such study is the the surface kinetics of CO oxidation catalysis at RuO_2(100) [2], where clarifying the system evolution contributed to the understanding of the catalytic reaction conditions. Besides KMC, methods like *Molecular Dynamics* (MD) could also be used in similar cases, e.g. in examining the temporal evolution of average atomic displacement over time, of a structure formed by Si atoms [4]. However, one of serious problem of MD is its inability to deal with larger scales. Typical steps of time are restricted within ∼ns, otherwise "artifacts" would

*Email: hzhaoaz@connect.ust.hk

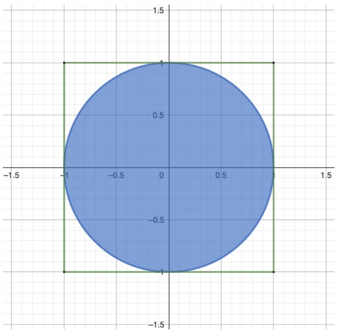

Figure 1: The MC example setup: unit circle and its circumscribed square. We throw N random darts on the square, and record the number n of those landing in the circle. $\frac{n}{N}$ provides an estimation to the area ratio, leading to an numerical solution of π.

N	π Estimation
10	2.8
10^2	2.76
10^6	3.142076
10^8 (1st try)	3.1414174
10^8 (2nd try)	3.1415044
10^8 (3rd try)	3.14162332
10^{10}	3.1415993184

Table 1: Python simulation. Increasing dart number N gives satisfactory estimations

be present in results due to the numerical *discretization error* [5, 6, 7]. KMC, on the other hand, has timescales from $\sim$ns to $\sim$s due to its hopping feature that utilizes the natural time separation of *rare events* [5, 8]. Therefore, speaking of studying particle systems in non-equilibrium, KMC is a method of its own unique value.

There have been many reviews on KMC ([5, 9, 10, 11] etc.). We would explore the the basic theory and algorithm of KMC in the Section 2. Based on the gained understanding of KMC, in Section 3 we look into a lattice KMC system of square molecules, and examine the result of the simulations when we input large number of molecules.

2 Theories and Algorithm of KMC

2.1 The Theories of KMC

The theories of KMC are based on two main principles: *rare event dynamics* and the *memorylessness property*. Besides these basics, in Section 2.3.1 cases are further simplified with the *lattice* assumption.

2.1.1 Rare Event Dynamics

Elementary processes of a system could be roughly categorized into 2 types: those with high activation barrier ($\gg k_BT$) and other vibrational motions. The former, having a low possibility of happening, is termed *rare event*. In terms of the potential energy surface (PES) and states of the system, the latter motions are around a single minimum in PES, being in the same state, while rare events are those transitioning among stable states and from one PES basin to another.

Timescale-wise, the time between rare events are of many orders of magnitudes longer (scale of $\sim\mu$s to $\sim$s compared with scale of $\sim$ps). Thus on a mesoscopic scale, it is reasonable to neglect the vibrational processes around one PES basin, and reprsent the ensemble of them with the state in that basin. This is called *coarse graining*. This gives us a simplified mesoscopic picture of systems in non-equilibrium: either being in metastable states, where most of the time is spent, or *hopping* between these states.

Now we put the picture in motion. The system is propagating between metastable states and hops among them, until a final steady state, if possible, is reached, i.e. State 1 $\xrightarrow{\text{hop}}$ State 2 $\xrightarrow{\text{hop}}$ ... $\xrightarrow{\text{hop}}$ Steady state.

2.1.2 The Memorylessness Property

As discussed above, the time within a metastable state is significantly longer than the hopping processes that brings the system in or out of it, thus the system presents a *memorylessness property*, both *within* and *among* each metastable state. The memorylessness property combined with the chain structure in Section 2.1.1 gives us the *Markov chain*.

KMC simulations are based on the Markov chain, and, with the right algorithm and input, could produce results close to direct Molecular Dynamics (MD) simulations, with far less compute workload.

Memorylessness among Metastable States KMC believes that the possibility $P_i(t)$ of the system to be in one state i, is only dependent on *two* probabilities: the probabilities of hopping out of state i into any other state j, k_{ij}, and on the probabilities of hopping into i from any other state j, k_{ji}. Here the probabilities are represented by rate constants of the hopping process, with units time^{-1}. Thus $P_i(t)$ obeys a simple balancing equation – the Markovian Master Equation:

$$\frac{\mathrm{d}P_i(t)}{\mathrm{d}t} = -\sum_{j\neq i} k_{ij}P_i(t) + \sum_{j\neq i} k_{ji}P_j(t). \tag{1}$$

Memorylessness within a Metastable State When the system is trapped in one potential basin, zooming in again, we would find the vibrations random and unconnected, as the drastic difference in timescale of one single vibration and the whole time span trapped in the state, say state i. As a result, the conditional probability distribution of the system escaping state i is irrelevant to the time our examination to the system (in state i) begins. Denoting the probability of escaping i as P_i^e, mathematically:

$$\forall \Delta t \in R^+, \forall r \in R^+,\ P_i^e(t > r + \Delta t \mid t > r) = P_i^e(t > \Delta t)$$

Notice that for the exponential function $f(x) = k'e^{-kx}$ (k, k' constant), $\forall r \in R^+$ and $\forall \Delta x \in R^+$:

$$\frac{\int_{r+\Delta x}^{\infty} f(x)}{\int_{r}^{\infty} f(x)} \equiv \frac{\int_{\Delta x}^{\infty} f(x)}{\int_{0}^{\infty} f(x)} = e^{-k\Delta x}.$$

Thus the escape probability *density* function follows the form of $f(x)$ (see Figure 2), leading to the escape probability of state i expression:

$$p_{ij} = k_{ij}e^{-k_{ij}t},\ p_{\text{tot}} = k_{\text{tot}}e^{-k_{ij}t}, \tag{2}$$

where p_{ij} is for escaping to state j, and p_{tot} is for all possible escape path.

Besides, there is:

$$k_{\text{tot}} = \sum_{j=1}^{M} k_{ij}. \tag{3}$$

Equation (2) is useful when weighting the system time advancement, which is covered in Section 2.2.

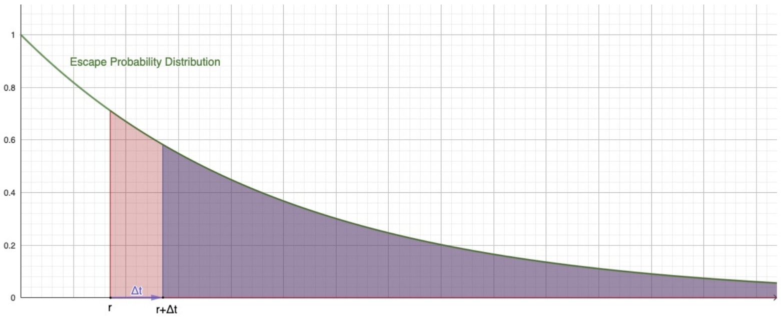

Figure 2: The memorylessness property of the escape probability *distribution* function. As $r \in R^+$ varies, with fixed Δt, the ratio of the 2 shaded areas is constant.

2.1.3 The Necessity of A Numerical Approach

Under the assumption of the Markov chain, we arrived at the two relations (1) and (2). Though it seems tempting to analytically solve $P_i(t)$ for any state i, it is in fact impossible, as the mere storage of the matrix k_{ij} $(\forall i, j)$ would be unfeasible. Take a 2D 300 × 300 lattice surface as an example, where each position corresponds to 2 states – occupied or not. The dimension of the matrix would be $2^{300 \cdot 300}$. Similar issues occur on our attempt for a *full* rate catalog in the KMC procedure, as is discussed in Section 2.3.1.

Thus we still need Section 2.2 to provide a numerical solution.

2.2 The Algorithm of KMC

We introduce the most commonly used KMC algorithm: the *n-fold way algorithm* (also known as the *BKL algorithm*, after the authors) [12], with given rate constants. Their calculations would be covered in Section 2.3. The algorithm utilizes 2 random numbers, ρ_1 and ρ_2, respectively in advancing the clock and determining the next state.

Advancing the Clock One feature of the n-fold way algorithm is that the escape time depends only on the total rate constant of the corresponding state, and not on the actual next state decided below. Recall the escape probability distribution p_{tot} in Equation (2). Our time advancement Δt would be weighted by the distribution, instead of following the straightforward relation $\Delta t = \frac{1}{k_{\text{tot}}}$. The weighting is implemented by drawing a random number $\rho_1 \in (0, 1]$ in this way:

$$\Delta t = -\frac{\ln(\rho_1)}{k_{\text{tot}}}. \tag{4}$$

where $\ln$ being the natural logarithm.

Deciding the Next State to Go Say our system sits in state i, with a total of M states possible next: state $1, 2, 3, ..., M$. In the n-fold way algorithm the next hopping destination is decided by a random selection of the possible states, and the randomness is achieved by the drawn number $\rho_2 \in [0, 1]$. Meanwhile, faster processes with higher rate constants are more probable to happen, thus the randomness is weighted as well, by the magnitude of the rate constants k_{ij} $(j \in \{1, 2, ..., M\})$. The weighting is implemented, as shown in Figure 3, by placing M blocks of width respectively proportional to k_{ij} $(j \in$

$\{1, 2, ..., M\}$) side by side on the x-axis, and see where the drawn point $(\rho_2 k_{\mathrm{tot}}, 0)$ lands in.

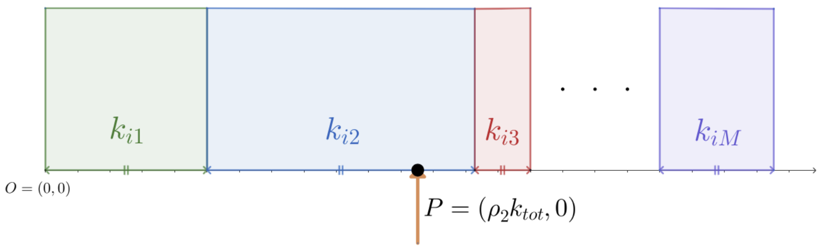

Figure 3: The weighted random selection algorithm for the next state. In this case the drawn point P lands in the region corresponding to state 2, thus the system would hop to that state.

After tackling the above two key problems, we have a clear picture of the n-fold way algorithm, as shown below in Figure 4.

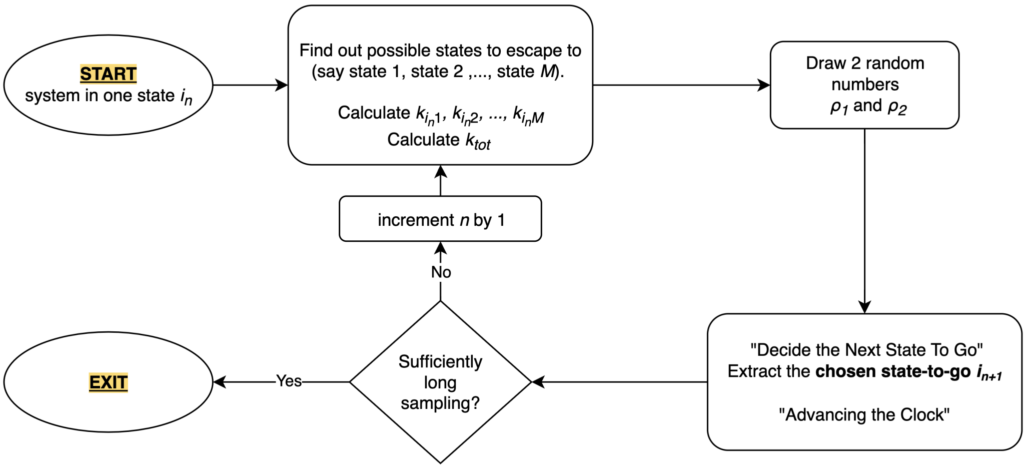
START
system in one state i_n
Find out possible states to escape to (say state 1, state 2 ,..., state M).
Calculate k_{i_n1}, k_{i_n2}, ..., k_{i_nM}
Calculate k_{tot}
Draw 2 random numbers ρ_1 and ρ_2
increment n by 1
No
"Decide the Next State To Go"
Extract the **chosen state-to-go i_{n+1}**
"Advancing the Clock"
Sufficiently long sampling?
Yes
EXIT

Figure 4: The n-fold way (BKL) algorithm flow chart. The ambiguity in ”sufficiently” results from the fact that KMC currently lacks a practical *steady-state detection* (SSD) method [9].

2.3 Determining The Rates

From Section 2.2, we could see that the original KMC algorithm requires *two* inputs: the possible states of the system, say state $1, 2, 3, ..., N$; and the matrix of rate constants $k_{ij}, \forall i, j \in \{1, 2, ..., N\}$.

Here we introduce how the rate constants are approximated using the commonly used *Transition State Theory* (TST), given the process (e.g. state $i \xrightarrow{\text{hop}}$ state j). Introducing the concept of *Transition State* (TS, i.e. state $i \xrightarrow[\text{via TS}]{\text{hop}}$ state j) and based on assumptions like crossing the energy barrier is classical and the hopping process is irreversible, TST deduces the elegant *Eyring Equation*:

$$k_{ij}^{\mathrm{TST}} = \frac{q_{\mathrm{TS}}^{\mathrm{vib}}}{q_i^{\mathrm{vib}}} \frac{k_{\mathrm{B}}T}{h} e^{-\frac{\Delta E_{ij}}{k_{\mathrm{B}}T}} = k_0 \frac{k_{\mathrm{B}}T}{h} e^{-\frac{\Delta E_{ij}}{k_{\mathrm{B}}T}} = f_{\mathrm{c}}\, e^{-\frac{\Delta E_{ij}}{k_{\mathrm{B}}T}}, \tag{5}$$

where $k_0 := \frac{q_{\mathrm{TS}}^{\mathrm{vib}}}{q_i^{\mathrm{vib}}}$. $k_0 \frac{k_{\mathrm{B}}T}{h}$ denoted as the *prefactor*, f_{c}; $q_{\mathrm{TS}}^{\mathrm{vib}} \leftrightarrow$ partition functions at the transition state; $q_i^{\mathrm{vib}} \leftrightarrow$ partition functions at the initial state; $k_{\mathrm{B}}T \leftrightarrow$ product of the Boltzmann constant and the absolute temperature; $h \leftrightarrow$ Planck's constant; $\Delta E_{ij} \leftrightarrow$ activation barrier of the state $i \xrightarrow[\text{via TS}]{\text{hop}}$ state j process.

In Equation (5), k_B, T, and h are constants. k_0 could be calculated in principle, one of the popular methods being *harmonic TST*. However, by experience, it is enough for most cases to approximate $k_0 \approx 1 - 10$, giving a prefactor f_c in the range $10^{12} - 10^{13}\ \mathrm{s}^{-1}$ [9].

Methods for calculating the activation barrier ΔE_{ij} is directly available from PES information and could be calculated first-order. Besides, for common species, relevant information (e.g., $\Delta_\mathrm{f} H^\circ$ and $\Delta_\mathrm{f} G^\circ$) could be found in databases like [13].

Before inputing the obtained rate constants, there is double-checking to do: the *Detailed Balance*. In short, there are two guidelines: first, for any $\text{state } i \xrightarrow[\text{via TS}]{\text{hop}} \text{state } j$, there must exist a reverse process: $\text{state } j \xrightarrow[\text{via TS}]{\text{hop}} \text{state } i$; Second, derived from the MME (1) in equilibrium, the following holds:

$$\frac{k_{ij}}{k_{ji}} = e^{-\frac{F_j(T) - F_i(T)}{k_\mathrm{B} T}}, \tag{6}$$

where $F_j(T) - F_i(T)$ is the free energy difference between the states.

2.3.1 Further Simplification: Mapping The System onto A Lattice

We further simplify the system by a stronger constraint – let the Markovian hops happen on a *lattice*, as is elaborated below in Figure 5.

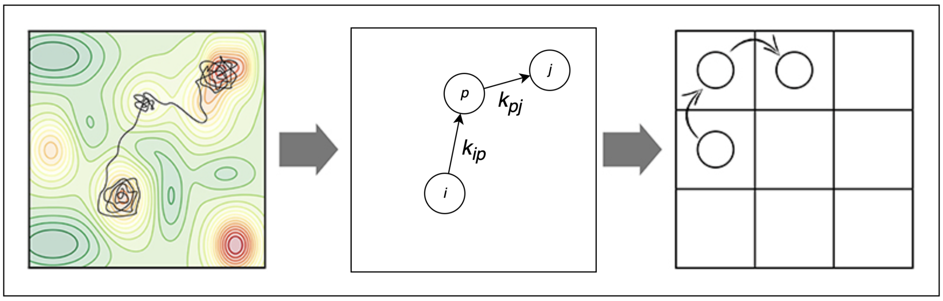

Figure 5: The completed coarse-graining process. **(Left, from [9])** A possible MD trajectory (black) overlayed on the underlying potential energy surface (PES) of the system with red regions representing lower-energy basins [9]. **(Center)** Coarse-graining of MD trajectory into a Markov chain. **(Right, from [9])** Coarse-graining of mapping the PES basins on suitable positions of the lattice.

The main challenge of the n-fold way algorithm is that it needs a gigantic database of *every* process and their corresponding rates (a *rate catalog*), which is generally not feasible to construct [9]. The benefit of the lattice mapping lies in simpler rate constant calculations, as the geometry complexity is greatly reduced.

Simpler geometry reduces the number of possible states and in turn the size of the rate catalog. Furthermore, it creates more equivalent processes. As a result, we could use one rate constant for multiple symmetric paths, like one atom hopping to 4 adjacent positions on an empty lattice; Besides, it is also more efficient when discarding processes, as they could be easily categorized. For instance, consider a 2-atom system, and we are constructing the rate catalog for one of the atoms. We could quickly categorize the paths $\text{state } i \xrightarrow[\text{via TS}]{\text{hop}} \text{state } j$ ($\forall i, \forall j$ in the rate catalog) by the distance of the 2 atoms in state i, and not consider those with relatively large distance, say those longer than 2 blocks. Thus the actual number of rate constant calculations needed (but not the size of the catalog) is decreased.

Lateral Interactions The above argument reveal one topic of lattice KMC worth consideration – lateral interactions: interactions between species adsorbed to a lattice. Thanks to the ease of pathway categorization, lateral interactions could be accounted for the *general cluster expansion method*, where the lattice energy is expanded into the sum of discrete interactions (*clusters*), like pairwise interactions, three-body interactions etc.. Each interaction is again broken down by distance: being nearest-neighbor, being next nearest neighbor etc.. Generally considering pairwise interactions is sufficient.

2.4 A Brief Deeper Dive into The Field of KMC

Above we covered the fundamental theories and algorithm of KMC, and how to determine the rates. Besides the basics above, KMC is an active field with numerous results on more advanced topics and new variations of the original algorithm, which we would not cover in detail.

Such topics include the *Garbage In Garbage Out (GIGO) Principle* [9]. It states that the quality of the input data would greatly impact the quality of the KMC results, and that KMC is fundamentally inexact for it is impossible to eliminate the ugly possibility of having unexpected pathways, leading to an imprecise rate catalog. Other topics include *Sensitivity Analysis*, determining the *true* steady state (mentioned in Figure 4), and the *time scale disparity problem* [9].

Examples of the new variations include *off-lattice KMC* and *on-the-fly* KMC [14].

3 The Square Lattice KMC System

In this section we introduce a lattice KMC system of cross-shaped molecules, implemented by GTK-Fortran [15, 16]. The system obeys the two principles of KMC (see Section 2.1) and follows the n-fold way algorithm (see Figure 4). Thus to unravel its mechanism, we only need to first understand the system setup (in Section 3.1) and then map the standard algorithm to our case (in Section 3.2), i.e. clarify the way we deduce the pathways and corresponding rate constants from the setup. This is equivalent to the construction of the rate catalog and thus the successful implementation of the system, with a random initialization algorithm of molecules at the beginning of simulations. Finally in Section 3.3, we perform simulations on the system.

3.1 System Setup

The system consist of cross-shaped molecules, and its lattice ”background”. Their properties could be modified by user inputs.

3.1.1 The Lattice

The lattice has the following properties (see Figure 6):

1. Its cells are hexagonal, i.e. each site having 6 adjacent sites.

2. It is 2D, with equal dimensions.

3. It is finite, yet translational symmetry is kept, as species extending out of one side would appear on the opposite side.

3.1.2 The Molecules

Static Geometric Properties As shown in Figure 6, the geometry of the molecule is that it has four arms extending out of a central atom. Each arm is of the length of 2 lattice sites and its end could form a bond with one arm of another molecule. Given that they live on lattice with *hexagonal* cells, the molecules could form two kinds of islands – diamond and Kagome.

Properties in Motion For state hopping of a molecule, two kinds of pathways could occur – *translation* and *rotation*. The pathways are crossing three types of *activation barriers* (recall TST and Equation (5)): *diffusion energy* E_d, *rotational energy* E_r, and intermolecular *bonding energy* E_b. Translation is associated with E_d and E_b. Rotation is associated with E_r and E_b.

Now we briefly explore the origin of the three kinds of energy barriers by investigating the interactions on lattice.

One molecule on lattice interact with two kinds of objects, the lattice itself and other molecules. First we discuss lateral interactions (Section 2.3.1). We employ the cluster expansion method discussed Section 2.3.1, i.e. creating a categorization of the lateral interactions based on how many molecules are involved and how far they are from one another. Everything is discarded except the most basic kind: those being pairwise and between adjacent molecules, just acting like a short-range *bond*. Therefore, the corresponding energy is vividly termed bonding energy E_b. Symmetry lead to the same value of E_b for every bond on every molecule, and the total barrier being $n_b E_\mathrm{b}$ for the case of n_b bonds on one molecule.

The origin of E_r and E_d is more straightforward as they correspond to the two kinds of hopping pathways. The simplification is assuming spacial symmetry and thus the equivalence of the molecule-lattice interactions across the lattice. As a result, there is only one value of E_d and E_r for any molecule in any situation.

In conclusion, we have two assumptions:

- molecular interactions being pairwise and between adjacent molecules
- the spacial symmetry of the molecular-lattice interactions.

Based on them, for every pathway of every molecule, there is only one mutual value of E_d and E_r (denoted E_{d_0} and E_{r_0}, respectively), and at most five values of E_{b_0}: $n_b E_\mathrm{b}$ ($n_b = 0, 1, 2, 3, 4$), where E_{b_0} is the value with one bond .

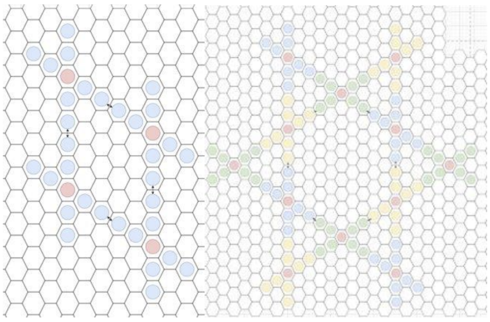

Figure 6: Figure from [15]. The geometry of the lattice plane and that of the molecules determine that the islands could only have two kinds – **(Left)** *diamond* island; **(Right)** *Kagome* island.

3.1.3 User Input

In the system, the user never directly edit the rate catalog. They input other parameters, and in this way change the automatically-generated rate catalog. These parameters include number of the lateral molecules n, the size of the plane *side*, temperature T, and the energies (bonding E_{b}, rotation E_{r}, and diffusion E_{d}). Total number of hopping of all molecules (termed *turns*) N_{tot} could also be modified to determine when to exit the simulation. This needs care so as to make sure saturation is reached in the system ("sufficiently long sampling" as in Figure 4).

3.2 Constructing The Rate Catalog

The Pathways In our system there are only *two kinds* of pathways due to symmetry and neglecting higher-order processes (e.g. transition to non-adjacent sites). They are rotation and transition to adjacent sites, as mentioned in Section 3.1.Considering the discussions in Section 3.1.2 and geometry, there are 10 rotation paths, as there are 2 rotational directions (restricted to only left and right, for simplicity) and each respectively having 5 bonding numbers n_{b}. Note that only 5 is worth considering due to spacial symmetry. Similarly, there are 5 unique transition paths as well.

The Rate Constant Recall Equation (5). We take approximation of the prefactor f_{c} as discussed in Section 2.3, and thus only need to determine the activation barrier of the processes ΔE_{ij}. Based on the above discussions, for individual transitional and rotational pathways T and R, respectively:

$$\Delta E_{\mathrm{T}} = E_{\mathrm{d}_0} + n_{\mathrm{b}} E_{\mathrm{b}_0},\ \Delta E_{\mathrm{R}} = E_{\mathrm{r}_0} + n_{\mathrm{b}} E_{\mathrm{b}_0}. \tag{7}$$

Based on Equation (5), for these pathways, the rate constants K_T and K_R are respectively:

$$k_{\mathrm{T}}^{\mathrm{TST}} = f_{\mathrm{c}}\, e^{-\frac{E_{\mathrm{d}_0} + n_{\mathrm{b}} E_{\mathrm{b}_0}}{k_{\mathrm{B}} T}},\ k_{\mathrm{R}}^{\mathrm{TST}} = f_{\mathrm{c}}\, e^{-\frac{E_{\mathrm{r}_0} + n_{\mathrm{b}} E_{\mathrm{b}_0}}{k_{\mathrm{B}} T}}. \tag{8}$$

n_{b} fixed for the system in a particular state. Thus from 3 and the symmetry discussed above, in this state we have:

$$k_{\mathrm{tot}} = n_{\mathrm{t}} k_{\mathrm{T}}^{\mathrm{TST}} + n_{\mathrm{r}} k_{\mathrm{R}}^{\mathrm{TST}}, \tag{9}$$

where n_{t} and n_{r} respectively denote the possible translation and rotation processes, during which the rate constants remain unchanged.

3.3 Running Simulations: Large Number of Molecules

Considering that [15] does not cover the results when the number of atoms present in the lattice are relatively large, here we examine the system with user input of 1500 and 2000 molecules, under different temperatures. Other inputs of the system are constant across simulations: $side = 300, E_{\mathrm{b}} = 0.1100, E_{\mathrm{r}} = 0.0040, E_{\mathrm{d}} = 0.0015$, and $N_{tot} = 10^9$.

As in Figure 8, starting from random positioned molecules, we run simulation of the systems and discuss the eventual preference of structure, i.e. Kagome-winning, diamond-winning, or coexistence. Principle for such judgement is whether the largest island size of the structure exceed a certain boundary. If only one structure does, that structure wins; otherwise it is considered coexistence. Setting this boundary is quite experience-based. For the 250K $n = 600$ case, for instance, the boundary is set to be 700 molecules. As shown in Figure 8 in there is a case of coexistence with each having a largest island of $\sim$ 600 molecules and there is a case of Kagome winning with a $\sim$ 800 molecules largest island. Note that the boundary varies among setups. The simulation result is shown in Table 2 below.

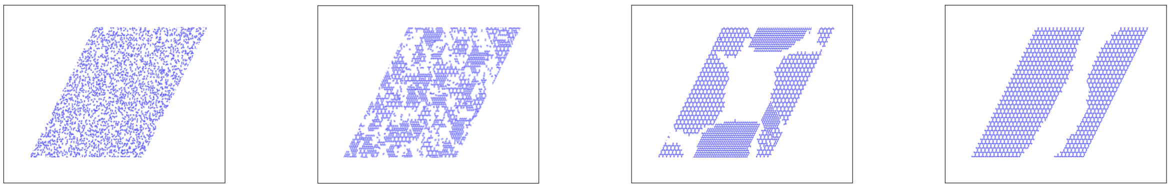

Figure 7: Transformation from randomness to order. Two figures on the left side are from the very beginning of simulations, while the two on the right side are the ordered structures in the very end. The far left being a Kagome-winning case, while the middle one being a coexistence case.

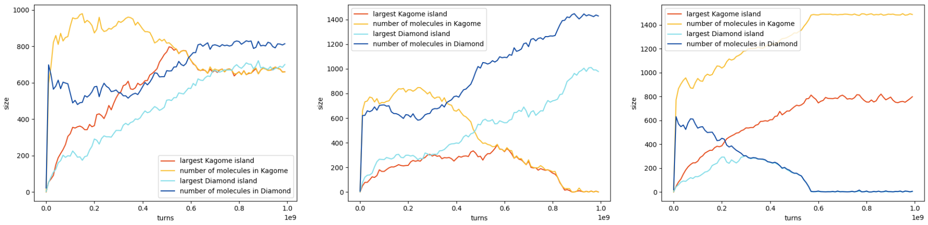

Figure 8: Determining the boundary. Determining the boundary first by experience, then it could be integrated in programs for automation. **(Left)** a coexistence case. **(Middle)** the diamond-winning case with more than one major diamond islands **(Right)** the Kagome-winning case with more than one major Kagome islands.

T	Number of Simulations	Kagome-win (with %)	Diamond-win (with %)	Coexist (with %)
150K ($n = 2000$)	10	0	$2 \leftrightarrow 20\%$	$8 \leftrightarrow 80\%$
250K ($n = 2000$)	9	$5 \leftrightarrow 55.6\%$	$2 \leftrightarrow 22.2\%$	$2 \leftrightarrow 22.2\%$
300K ($n = 2000$)	10	$7 \leftrightarrow 70\%$	$3 \leftrightarrow 30\%$	0
350K ($n = 2000$)	10	$9 \leftrightarrow 90\%$	$1 \leftrightarrow 10\%$	0
250K ($n = 1500$)	10	$8 \leftrightarrow 80\%$	$2 \leftrightarrow 20\%$	0
300K ($n = 1500$)	9	$7 \leftrightarrow 77.8\%$	$1 \leftrightarrow 11.1\%$	$1 \leftrightarrow 11.1\%$

Table 2: Results of $n = 1500$ and $n = 2000$ setups under different temperatures.

We could deduce several findings based on the results above. It could be further verified by future studies.

1. Overall Kagome is far more likely than other two cases. Yet notice when $n = 200$ there is a sudden drop of Kagome winning cases between 150K and 250K. Smaller temperature steps could be included to examine under which temperature this drop happened.

2. When comparing 1500K groups with 2000 groups, we could see that enlarging number of molecules has the same effect of decreasing the size of the plane, as mentioned in [15]. The portion of diamond-winning case grew larger as n increases.

3. Larger n requires longer simulation time. As shown in [15], $N_{\text{tot}} = 10^9$ is well enough for $n = 400$. However in our simulation there are systems that are clearly unsaturated yet, for example the case in Figure 9.

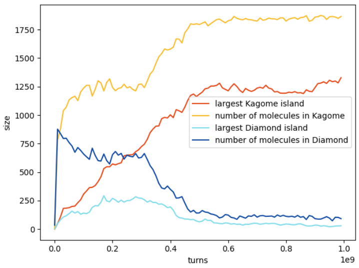

Figure 9: Unsaturated system case. The trend of the largest Kagome island is ambiguous.

4 Conclusion

In this report, we covered the fundamentals of the *kinetic Monte Carlo* (KMC) method, and have put it in the context of an example lattice system of cross shape molecules. Then we run simulations with input of large number of molecules, and studied the results.

In the introduction, KMC's flexible timescale advantage over other methods like *Molecular Dynamics* was introduced, and thus it has a wide range of use in active areas requiring the understanding of the *dynamical* evolution of systems. Then the journey of understanding KMC begins. We started from the central yet simple idea of all *Monte Carlo* simulations of using random numbers. We later understood that KMC utilizes two random numbers, and weighting is involved in both of the processes – using the random number to exponentially weight the time increment after one transition, and randomly select the state-to-go with the probabilities weighted with rate constants. From the two basic modelizations of *rare event dynamics* and the *memorylessness* property, we arrived at the *Markov chain*, a key concept of the KMC setup. Based on the assumptions we also deduced the *Markovian Master Equation* (MME) and the exponential decay of the escape probability over time. Yet despite the theoretical possibility of an analytical approach, going numerical was necessary because of the almost-impossible calculations. The *n-fold way* algorithm was introduced as the most classical KMC algorithm to offer numerical insight into the MME. It consists of constructing the rate catalog, drawing the random numbers, advancing the clock, and decide the next state to go based on the weighted probability, as discussed above. The concise yet powerful loop continues until, by experience, we acknowledge it is sufficient to exit. Then we introduced the *Transition State Theory* and the Eyring Equation that offered the approximate of rate constants. The determined rates together with the lattice assumption made the calculations affordable and thus the algorithm practical.

In the following section we implemented the above understandings into the investigation of one particular system. From scratch, we walked through the implementation of the n-fold way algorithm in this special case, from understanding the system setup, to making practical assumptions for simplicity, to eventually deducing the rate constants. Then we run simulations on the system after understanding its mechanism, looking into the case of large number of molecules.

We realize that the understandings and results of this report are elementary. In the future, based on these fundamental interpretations, deeper theoretical foundations of KMC could be explored, and simulations of various setup and input could be conducted.

5 Acknowledgements

I would like to appreciate the guidance of Prof. Lin and senior undergraduate student Alpha Hui. Alpha generously shared his past UROP reports and the cross-shape molecule system. He also patiently answered my questions. Prof. Lin as well as Alpha kindly listened to my presentations on weekly meetings and offered valuable insight. This UROP tasting stream experience would be much more challenging and less rewarding without their help.

References

[1] Voter, A. F. (1986, November 15). Classically exact overlayer dynamics: Diffusion of rhodium clusters on Rh(100). *Physical Review B, 34(10), 6819–6829.* `https://doi.org/10.1103/physrevb.34.6819`

[2] Reuter, K., Frenkel, D., and Scheffler, M. (2004, September 10). The Steady State of Heterogeneous Catalysis, Studied by First-Principles Statistical Mechanics. *Physical Review Letters, 93(11).* `https://doi.org/10.1103/physrevlett.93.116105`

[3] Kotrla, M. (1996, August). Numerical simulations in the theory of crystal growth. *Computer Physics Communications, 97(1–2), 82–100.* `https://doi.org/10.1016/0010-4655(96)00023-9`

[4] Car, R., and Parrinello, M. (1985, November 25). Unified Approach for Molecular Dynamics and Density-Functional Theory. *Physical Review Letters, 55(22), 2471–2474.* `https://doi.org/10.1103/physrevlett.55.2471`

[5] Kratzer. (2009, April). Monte Carlo and kinetic Monte Carlo methods. *ArXiv.* `https://doi.org/10.48550/arXiv.0904.2556`

[6] Davidchack, R. L. (2010, December). Discretization errors in molecular dynamics simulations with deterministic and stochastic thermostats. *Journal of Computational Physics, 229(24), 9323–9346.* `https://doi.org/10.1016/j.jcp.2010.09.004`

[7] Jakobsen, A. F., Mouritsen, O. G., and Besold, G. (2005, May 20). Artifacts in dynamical simulations of coarse-grained model lipid bilayers. *The Journal of Chemical Physics, 122(20).* `https://doi.org/10.1063/1.1900725`

[8] Henkelman, G., and Jónsson, H. (2003, March 21). Multiple Time Scale Simulations of Metal Crystal Growth Reveal the Importance of Multiatom Surface Processes. *Physical Review Letters, 90(11).* `https://doi.org/10.1103/physrevlett.90.116101`

[9] Andersen, M., Panosetti, C., and Reuter, K. (2019, April 9). A Practical Guide to Surface Kinetic Monte Carlo Simulations. *Frontiers in Chemistry, 7.* `https://doi.org/10.3389/fchem.2019.00202`

[10] Voter, A. F. (2007). INTRODUCTION TO THE KINETIC MONTE CARLO METHOD. *Radiation Effects in Solids, 1–23.* `https://doi.org/10.1007/978-1-4020-5295-8_1`

[11] Battaile, C. C. (2008, July). The Kinetic Monte Carlo method: Foundation, implementation, and application. *Computer Methods in Applied Mechanics and Engineering, 197(41–42), 3386–3398.* `https://doi.org/10.1016/j.cma.2008.03.010`

[12] Bortz, A., Kalos, M., and Lebowitz, J. (1975, January). A new algorithm for Monte Carlo simulation of Ising spin systems. *Journal of Computational Physics, 17(1), 10–18.* `https://doi.org/10.1016/0021-9991(75)90060-1`

[13] JANAF THERMOCHEMICAL TABLES. (1989, December 1). *Analytical Chemistry, 61(23), 1327A-1327A.* `https://doi.org/10.1021/ac00198a726`

[14] Henkelman, G., and Jónsson, H. (2001, December 1). Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *The Journal of Chemical Physics, 115(21), 9657–9666.* `https://doi.org/10.1063/1.1415500`

[15] Hui, L.H.A. (n.d.). UROP4100 Final Report: Kinetic Monte Carlo Simulation of 2D supramolecular assembly. *HKUST UROP.*

[16] Magnin, V., Tappin, J., Hunger, J., and De Lisle, J. (2019, February 12). gtk-fortran: a GTK+ binding to build Graphical User Interfaces in Fortran. *Journal of Open Source Software, 4(34), 1109.* `https://doi.org/10.21105/joss.01109`